IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application:
Vyacheslav Barsuk
Serial No.: 10/736,429
Filed: 12/15/2003
Title: METHOD, APPARATUS & PROGRAM STORAGE DEVICE FOR PROVIDING A REMOTE POWER RESET AT A REMOTE SERVER THROUGH A NETWORK CONNECTION

Group Art Unit: 2116
Examiner: Michael J. Brown
IBM Corporation
Intellectual Property Law
Department IQ0A/040-3
1701 North Street
Endicott, NY 13760

Commissioner For Patents
PO Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

I. Real Party in Interest

International Business Machines Corporation is the real party in interest.

II. Related Appeals and Interferences

There are no related appeals or interferences or other judicial proceedings.

III. Status of Claims

Claims 1-8, 10-15, 18 and 21 are pending, Finally Rejected and Appealed.

Claims 9, 16-17, 19-20 and 22 have been Canceled.

IV. Status of Amendments

On November 22, 2006, after Final Rejection, Appellant filed an Amendment under Rule 116 which was not entered because of lack of clarity as to which claims were canceled. On January 17, 2007, Appellant filed another Amendment under Rule 116 which clearly indicated which claims were canceled, and this Amendment under Rule 116 was entered.

On June 26, before Final Action, Appellant filed a Rule 131 Affidavit including original source code and an original directory to swear behind US 2004/0267918 to Guarraci et al. The Examiner entered this Rule 131 Affidavit, but could not understand the source code and therefore, stated that it was not effective to swear behind Guarraci et al. On November 22, 2006, after Final Rejection, Appellant filed a Supplemented Rule 131 Affidavit to swear behind Guarraci et al. The Supplemental Rule 131 Affidavit included the original source code and the original directory as previously filed by Appellants on June 26, 2006 before Final Rejection. The Supplemental Rule 131 Affidavit also included newly created explanatory comments prepared by the Inventor at approximately the time of filing the Supplemental Rule 131 Affidavit to assist the Examiner in understanding the source code. The Examiner did not enter Appellant's Supplemental Rule 131 Affidavit because the Examiner could not readily differentiate between the newly created explanatory comments to the source code, and the original source code, although Appellant believes that he furnished the explanatory comments in a distinguishing blue color. Also, the Examiner could have compared the original Rule 131 Affidavit (which did not include the explanatory comments) to the Supplemental Rule 131 Affidavit (which included the explanatory comments) to identify the newly created explanatory comments. The Examiner also did not enter the Supplemental Rule 131 Affidavit because the Examiner thought it was not seasonably presented under MPEP 715.09, even though the operative part of the Supplemental Rule 131 Affidavit was submitted before Final Rejection in the original Rule 131 Affidavit.

On January 15, 2007, Appellant resubmitted the same Supplemental Rule 131 Affidavit as previously filed on November 22, 2006, but the Examiner again refused to enter the Supplemental Rule 131 Affidavit, despite Appellant's arguments that the operative part of the

Supplemental Rule 131 Affidavit, i.e. the source code itself, was sufficient to swear behind Guarraci et al.

V. Summary of Claimed Subject Matter

Support for the elements in the claims is indicated in plain brackets [ ].

Claim 1 recites a method for performing a remote power reset at a remote server through a network connection. [Page 5 lines 7-16] A power reset procedure is pinned to memory at a remote server. [Page 3 lines 7-12. Figure 4 step 410. Page 9 lines 9-20. Page 12 lines 16-22. Remote Process 140 and Memory 132 of Figure 1.] The power reset procedure is continuously run to listen for a call to initiate a power reset in response thereto. [Page 9 line 20 - Page 10 line 4.]

Claim 10 recites a remote server comprising a memory for storing program instructions. [Page 5 lines 7-16] A power reset procedure is pinned to the memory. [Page 3 lines 7-12. Figure 4 step 410. Page 9 lines 9-20. Page 12 lines 16-22. Remote Process 140 and Memory 132 of Figure 1.] The remote server also comprises a processor configured according to the program instructions for running the power reset procedure to listen for a call to initiate a power reset in response thereto. [Page 9 line 20 - Page 10 line 4.]

Claim 18 recites a remote server with means for storing program instructions. [Page 5 lines 7-16, Memory 132 of Figure 1.] A power reset procedure is pinned to a memory. [Page 3 lines 7-12. Figure 4 step 410. Page 9 lines 9-20. Page 12 lines 16-22. Remote Process 140 and Memory 132 of Figure 1.] There are means configured according to the program instructions for running the power reset procedure to listen for a call to initiate a power reset in response thereto. [Page 9 line 20 - Page 10 line 4.]

Claim 21 recites a program storage device readable by a computer. [Page 13 line 20 to Page 14 line 5.] The program storage device tangibly embodies one or more programs of instructions executable by the computer to perform a method for performing a remote power reset at a remote server through a network connection. [Figure 4 step 410. Page 9 lines 9-20. Page 12 lines 16-22. Network 116 of Figure 1] According to the method, a power reset procedure is pinned to memory at a remote server. [Page 3 lines 7-12. Figure 4 step 410. Page 9 lines 9-20. Page 12 lines 16-22. Remote Process 140 and Memory 132 of Figure 1.] Also, the power reset procedure is continuously run to listen for a call to initiate a power reset in response thereto. [Page 9 line 20 - Page 10 line 4.]

The structure, material or acts described in the specification plus equivalents for each means plus function and step plus function are identified in stylized brackets { }.

Claim 10. A remote server, comprising:

a memory for storing program instructions {Memory 132 of Figure 1}, a power reset procedure {Remote Process 140} being pinned to the memory {Page 9 lines 11-12}; and

a processor {Processor 210 of Figure 2} configured according to the program instructions for running the power reset procedure to listen for a call to initiate a power reset in response thereto {Page 9 lines 20 - 22.}.

Claim 18. A remote server, comprising:

means for storing program instructions {Memory 132 of Figure 1}, a power reset procedure {Remote Process 140} being pinned to a memory {Page 9 lines 11-12}; and

means configured according to the program instructions for running the power reset procedure to listen for a call to initiate a power reset in response thereto {Page 9 lines 20 - 22.}.

VI. Grounds of Rejection to be Reviewed on Appeal

Claims 1-8, 10-15, 18 and 21 were rejected under 35 USC 102(e) as being anticipated by Guarraci et al. (US Pub 2004/0267918).

In an Advisory Action of January 4, 2007, the Examiner cited Gomi et al. as teaching "that pinning a program is quite well known in the prior art". However, the Examiner did not make a formal rejection based on Gomi et al. (US 6,301,634). Nevertheless, Appellant assumes that the Examiner rejected claims 1-8, 10-15, 18 and 21 under 35 USC 103 based on Guarraci et al. in combination with Gomi et al.

VII. Argument

Rejection of Claims 1-8, 10-15, 18 and 21 under 35 USC 102 based on Guarraci et al.

Claims 1-8, 10-15, 18 and 21 were rejected under 35 USC 102(e) as being anticipated by Guarraci et al. (US Pub 2004/0267918). Appellant respectfully traverses this rejection based on the following.

Explanation of Original Rule 131 Affidavit and **How it Swears Behind Guarraci et al.**

Guaracci et al. was filed on June 30, 2003. Appellant's original Rule 131 Affidavit (seasonably filed and entered before Final Rejection) included copies of "res.c" source code listing, "rescl.c" source code listing and "dir.list.txt". The res.c program was implemented at the client computer to request the power reset of the remote server. The rescl.c program was implemented at the remote server to pin memory for the power reset function, wait for the request for power reset and then initiate the power reset of the remote server. The directory shows the last dates of update of "res.c" source code and "rescl.c" source code, to be in March 2003. As indicated by the original Rule 131 Affidavit (including the source code listings) and Appellant's Invention Disclosure submitted on May 7, 2003, Appellant conceived and reduced to practice his invention, as currently claimed, by March 2003 which is before the filing date of Guaracci et al. The Examiner objected to the source code listings in the original Rule 131 Affidavit because they were not clear enough to the Examiner as to the function implemented by the two source code programs. However, this does not diminish the effect of the source code programs for swearing behind Guarraci et al. **Source code is readily understood by someone**

of ordinary skill in the art, and therefore, meets the requirements for proof of conception and reduction to practice. Moreover, the source code could readily be compiled into object code and executed, by a mere technician.

As a convenience to the Examiner, Appellant subsequently submitted a Supplemental Rule 131 Affidavit with the same "res.c" source code and "rescl.c" source code as in the original rule 131 Affidavit but with Appellant's newly created/added comments (in blue) next to key lines of the source codes to explain their meaning. (The Supplemental Rule 131 Affidavit is also enclosed for the convenience of the Board, with the newly created/added comments being underlined.) Appellant also enclosed another Affidavit swearing that Appellant's added comments were accurate. Regardless of whether the Supplemental Rule 131 Affidavit is considered, the following explanation applies to the original Rule 131 Affidavit which was entered. The last eight lines of the rescl.c source code, beginning with "if (getsockname ...", implements the pinning of the power reset program to memory, and the looping of the power reset program to listen for a call to initiate the power reset program, as recited in claim 1. The ninth line from the end of the source code binds the socket to the TCP port to listen for a TCP/IP request packet to perform power reset, as recited in claim 2 and elsewhere. The step "reboot (RB_SOFTIPL)", in about the middle of the source code, is a system call to reboot as recited in claim 7. In the res.c source code, the step "rc=connect ..." about in the middle of the source code, allows another computer to establish a TCP socket connection with the remote computer which includes the power reset program. The step "fpringf(stderr, "attempt to reboot/n" six lines from the end, establishes the request to reboot to be sent to the remote computer. Therefore, Appellant has established conception and actual reduction to practice of the invention, as currently claimed, prior to the effective date of Guaracci et al. Consequently, Guaracci et al. is not an effective reference against the present patent application.

<u>Substantive Rejection of Claims 1-8, 10-15, 18 and 21 under 35 USC 102 based on Guarraci et al.</u>

Claims 1-8, 10-15, 18 and 21 were rejected under 35 USC 102 based on Guarraci et al. Appellants traverse this rejection based on the following (in case the Rule 131 Affidavit is not found to be effective in swearing behind Guarraci et al).

Claim 1 recites the step of "pinning". "Pinning" is defined in the Background section of the present patent application as follows:

> "Pinning generally refers to an ability for pages to remain in main memory and not have to be swapped out, typically by a computer operating system. This enables memory pages to be maintained in real memory all the time. However, if a program/process is not pinned to memory (normally it is not), the program/process competes for memory resources with other programs. However, as soon as a new resource is required, e.g., memory, the program will fail." Present Patent Application, Page 3 lines 7-12.

Thus, "pinning" a program to memory means that the program will remain in memory for execution, and not be swapped out to storage, even when memory is in short supply, and another program may need memory.

Guaracci et al. fail to disclose this key features of the present invention, i.e. pinning a power reset procedure to memory at a remote server, and continuously running the power reset procedure to listen for a call to initiate a power reset in response thereto. The pinning enables the power reset procedure to be operative despite a shortage of memory at the remote server that causes other critical functions to fail. The pinning of the power reset procedure and the continuous running of the power reset procedure to listen for a call allows power reset of the remote computer to correct problems with the remote computer, when there is insufficient memory at the remote computer for all programs to run. Guaracci et al. are concerned with something different, i.e. a communication channel between a controlling computer and remote computers; and remotely monitoring computer systems over an out-of-band communication channel when the in-band communication channel is unavailable. Guaracci et al. does not teach that a power reset procedure is pinned to memory, so that it is not outpaged when memory is short. Moreover, Guaracci et al. do not teach continuously running the power reset procedure to listen for a call to initiate a power reset in response thereto. Therefore, even if Guaracci et al. were to predate the present invention, the rejection under 35 USC 102 should be withdrawn. Moreover, Guaracci et al. fail to teach or even suggest the present invention, so a rejection under 35 USC 103 would be equally unfounded.

The Examiner cites Paragraph 0035 of Guaracci et al., "UPS 120 may provide basic remote management capabilities, such as the ability to cycle power or reset headless server 110". This just means to turn the electrical power to server 110 on and off, like throwing a power switch. This does not involve pinning a power reset procedure to memory and then remotely calling the power reset procedure, as recited in claim 1.

The other independent claims 10, 18 and 21 similarly distinguish over Guarraci et al.

Assumed Rejection of Claims 1-8, 10-15, 18 and 21 under 35 USC 103 based on Guarraci et al. in view of Gomi

As explained above, Guarraci et al. fail to disclose the pinning of a power reset procedure to memory and continuously running the power reset procedure to listen for a call to initiate a power reset in response thereto, as recited in claim 1. The Examiner then cites Gomi to purportedly fill the gap. Gomi discloses that an "event registration unit registers the event object to be used during dummy booting in order to prevent swapping-out of an application program 236 that is waiting for the occurrence of an event." Gomi et al. attempt to keep an event registration application from being swapped-out of memory, because the event registration application is needed to respond in real time, and the reloading time would prevent a real time response by the event registration application. However, Gomi et al. do not teach "pinning" as recited in claim 1. Rather, Gomi et al. teach a swap-out prevention unit 90 which periodically invokes the event registration application for dummy operation. "The swap-out prevention unit 90 starts the application program on a periodic basis so that the application program will not be swapped out." See Column 12 line 41 to Column 13 line 16. Presumably, the swap-out algorithm in Gomi et al. is based on how long an application has not been used/invoked, so the periodic invocation of the event registration application by the swap-out prevention unit, even for dummy operation, convinces the swap-out algorithm not to swap-out the event registration application because there are less recently used applications to swap-out instead. The swap-out prevention unit of Gomi et al. does not supersede the swap-out algorithm, but instead makes the even registration application more recently used than other swap-out candidates. In contrast, claim 1 of the present patent application recites "pinning" which means that the program will maintain some place in memory to execute, and not be swapped out to storage, even though the memory is in short supply, and another program may need it. "Pinning" is a reservation for memory space (although not necessarily the same location in memory) and supersedes the normal swap-out algorithm. For example, because the restart program function is "pinned" to memory as recited in claim 1, a more recently used application will be swapped-out to storage when memory is in short supply instead of swapping-out the less recently used restart program function. Gomi et al. teach a different technique for preventing swap-out of an application. Therefore, Guarraci et al in combination with Gomi et al. do not teach or suggest the present invention as recited in claim 1. Neither teaches either element of claim 1, i.e. pinning of a power reset procedure to memory and continuously running the power reset procedure to listen for a call to initiate a power reset in response thereto.

The other independent 10, 18 and 21 similarly distinguish over Guarraci et al. in combination with Guarraci et al.

Appellant acknowledges that pinning other types of programs to memory is prior art. This is stated in the Background section of the present patent application. However, to pin a power reset program to memory and continuously run the power reset procedure to listen for a call to initiate a power reset in response thereto, as recited in claims 1, 10, 18 and 21 is new. It would not have been obvious to pin a power reset procedure to memory and continuously run the power reset procedure to listen for a call to initiate a power reset in response thereto, because this was not taught or suggested by the prior art. Also, this involves two discrete steps in combination to solve a problem with shortage of memory. The pinning of the remote power reset procedure to memory ensures that the power reset procedure will be in memory and callable despite the shortage of memory. Continuously running the power reset procedure to listen for a call to initiate a power reset in response thereto, avoids reliance on another "intervening" program (in the computer to be reset) to invoke the power reset procedure upon receipt of a remote power reset command. This other intervening program might be outpaged, and therefore not available when memory was short. So, the present invention advantageously does not include or use such an intervening program (in the computer to be reset). Therefore, claim 1 recites two discrete steps, in combination, that ensure operation of a power reset procedure during shortage of memory. The other independent 10, 18 and 21 similarly distinguish over the known technique of pinning other types of programs to memory.

Rejection of Claims 2-6, and 11-14 under 35 USC 102
based on Guarraci et al.

Dependent claims 2-6, and 11-14 recite that the call to the pinned, power reset procedure is made via a network. Guarraci et al. do not remotely call a pinned, power reset procedure.

Assumed Rejection of Claims 2-6, and 11-14 under 35 USC 103
based on Guarraci et al. and Gomi et al.

Dependent claims 2-6, and 11-14 recite that the call to the pinned, power reset procedure is made via a network. Neither Gurraci et al. nor Gomi et al. remotely call a pinned, power reset procedure or suggest such.

Rejection of Claim 8 under 35 USC 102
based on Guarraci et al.

Dependent claim 8 recites that the remote server hangs as result of running out of virtual memory, and (the step of) initiating the power reset further comprises (the step of) unhanging the remote server using the power reset procedure. Guarraci et al. do not teach or suggest a power reset procedure to correct a shortage of virtual memory.

Assumed Rejection of Claim 8 under 35 USC 103
based on Guarraci et al. and Gomi et al.

Dependent claim 8 recites that the remote server hangs as result of running out of virtual memory, and (the step of) initiating the power reset further comprises (the step of) unhanging the remote server using the power reset procedure. Neither Guarraci et al. nor Gomi et al. teach or suggest a power reset procedure to correct a shortage of virtual memory.

Based on the foregoing, Appellant requests that the Final Rejection of claims 1-8, 10-15, 18 and 21 be reversed.

Respectfully submitted,

Dated: 03/02/2007
Telephone: 607-429-4368
Fax No.: 607-429-4119

/Arthur J. Samodovitz/
Arthur J. Samodovitz
Reg. No. 31,297

VIII. CLAIMS INVOLVED IN APPEAL

1. A method for performing a remote power reset at a remote server through a network connection, comprising:

pinning a power reset procedure to memory at a remote server; and

continuously running the power reset procedure to listen for a call to initiate a power reset in response thereto.

2. The method of claim 1, wherein the listening for a call further comprises listening for a Request TCP/IP packet to initiate the power reset.

3. The method of claim 2, wherein the listening for a call further comprises listening for a call via a network connection.

4. The method of claim 1, wherein the listening for a call further comprises listening for a call via a network connection.

5. The method of claim 1, wherein the listening for a call further comprises listening to a predetermined TCP/IP port for a Request TCP/IP packet specifying the power reset procedure.

6. The method of claim 5 further comprising receiving at the predetermined TCP/IP port the Request TCP/IP packet and initiating the power reset procedure pinned in the memory of the remote server in response to receipt of the Request TCP/IP packet.

7. The method of claim 6, wherein the initiating the power reset further comprises performing a system call to reboot or power off the remote server.

8. The method of claim 6, wherein the remote server hangs as result of running out of virtual memory, and wherein the initiating the power reset further comprises unhanging the remote server using the power reset procedure.

10. A remote server, comprising:

a memory for storing program instructions, a power reset procedure being pinned to the memory; and

a processor configured according to the program instructions for running the power reset procedure to listen for a call to initiate a power reset in response thereto.

11. The remote server of claim 10, wherein the call comprises a Request TCP/IP packet for initiating the power reset.

12. The remote server of claim 10, wherein the processor listens for the call via a network connection.

13. The remote server of claim 10, wherein the processor listens for a call by listening to a predetermined TCP/IP port for a Request TCP/IP packet specifying the power reset procedure.

14. The remote server of claim 13, wherein the processor receives the Request TCP/IP packet from predetermined TCP/IP port and initiates the power reset procedure pinned in the memory in response to the Request TCP/IP packet.

15. The remote server of claim 14, wherein the processor initiates the power reset by performing a system call to reboot or power off the remote server.

18. A remote server, comprising:

means for storing program instructions, a power reset procedure being pinned to a memory; and

means configured according to the program instructions for running the power reset procedure to listen for a call to initiate a power reset in response thereto.

21. A program storage device readable by a computer, the program storage device tangibly embodying one or more programs of instructions executable by the computer to perform a method for performing a remote power reset at a remote server through a network connection, the method comprising:

pinning a power reset procedure to memory at a remote server; and continuously running the power reset procedure to listen for a call to initiate a power reset in response thereto.

IX Evidence Appendix

A copy of the original Rule 131 Affidavit which was submitted before Final Rejection and entered is enclosed. A copy of the Supplemental Rule 131 Affidavit which was submitted after Final rejection and not entered, is also enclosed for the convenience of the Board in understanding the source code contained in the original Rule 131 Affidavit. The Supplemental Rule 131 Affidavit which is enclosed has the same comments that were created at the time of filing the Supplemental Rule 131 Affidavit, but they are now underlined to standout better.

X. Related Proceedings Appendix

There are no related Appeals or other proceedings, and therefore no copies of decisions to include.